EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	562	lsync	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:44
S2	16	S1 and "711".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:45
S3	5	S1 and "712".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:45
S4	83	msync	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:45
S5	10	S4 and "711".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:47
S6	4	S4 and "712".clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:47
S7	8	S1 and S4	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:48
S8	5	vector and scalar and S1	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:48

EAST Search History

S9	4650	(prevent$4 block$4 stop$4) same scalar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:49
S10	1734	(prevent$4 block$4 stop$4) same scalar same vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:50
S11	113	(prevent$4 block$4 stop$4) same scalar same vector same cache	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:50
S12	90	S11 and ("711" "712").clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:50
S13	694	(scalar with (prevent$4 block$4 stop$4)) same vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:51
S14	63	(scalar with (prevent$4 block$4 stop$4)) same vector same cache	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:51
S15	6	(scalar with (prevent$4)) same vector same cache	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:52
S16	48	(scalar with (prevent$4)) same vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:53

S17	15	S16 and ("711" "712").clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:53
S18	0	(prevent$4 with access with scalar) same vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:54
S19	7	prevent$4 same access same scalar same vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 13:54
S21	97	vector same writ$4 same invalidat$4 same cache	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 14:02
S23	17	sync and S21	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 14:04
S24	43	synchroniz$5 and S21	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 14:04
S25	0	subsequent with prevent$4 with scalar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 14:11
S26	29	subsequent same prevent$4 same scalar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 14:11

EAST Search History

S27	318	vector with (before prior) with scalar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:20
S28	0	acess$4 with vector with (before prior) with scalar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:20
S29	0	acess$4 same (vector with (before prior) with scalar)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:20
S30	20	cache same (vector with (before prior) with scalar)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:21
S31	19	("3949379" | "4156906" | "4638431" | "4722049" | "4881168" | "4967350" | "5043886" | "5063497" | "5123095").PN. OR ("5247635").URPN.	US-PGPUB; USPAT; USOCR	OR	ON	2008/02/04 15:23
S32	18	("4244033" | "4638431" | "4942518" | "5038278" | "5123095" | "5247635" | "5333291" | "5335325" | "5353425" | "5418973" | "5426754" | "5430884").PN. OR ("5717895").URPN.	US-PGPUB; USPAT; USOCR	OR	ON	2008/02/04 15:26
S33	100	(memory near2 ordering) and vector and scalar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:30
S34	71	S33 and ("711" "712").clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:31

S35	30	synchron$5 and S33	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:31
S36	16	("20030167383"|"5375223"|"5530933"|"5796980"|"5835951"|"6014728"|"6519685"|"6987571").PN.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/04 15:57
S37	130	fence and scalar and vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:53
S38	35	S37 and ("711" "712").clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:54
S39	14	fence same scalar same vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:55
S40	7	membar and scalar and vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:56
S41	3173	membar	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:59
S42	70	S41 and ("711" "712").clas.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:59

S43	19	S42 and (scalar vector)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 14:59
S44	38	S41 and (scalar vector)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:00
S45	6	("5613083" | "5751983" | "5854914" | "5999727").PN. OR ("6430649").URPN.	US-PGPUB; USPAT; USOCR	OR	ON	2008/02/05 15:01
S46	55	membar and (invalidat$5 same cache)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:02
S47	0	membar same (invalidat$5 same cache)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:02
S48	50	membar and (invalidat$5 same cache) and synchroniz$5.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:03
S49	23	(memory adj barrier) and synchroniz$5 and scalar and vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:04
S50	23	(memory adj barrier) and scalar and vector	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:05
S51	69	(memory adj barrier) and synchroniz$5 and (scalar vector)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 15:05

EAST Search History

S52	37	(memory adj barrier) and sync and (scalar vector)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2008/02/05 21:51
S53	3511	(711/151,154,158 712/3,40,225). ccls.	USPAT	OR	ON	2008/02/05 21:52
S54	6	S53 and membar	USPAT	OR	ON	2008/02/05 21:52
S55	41	S53 and fence	USPAT	OR	ON	2008/02/05 21:53
S56	1	S53 and fence and vector and scalar	USPAT	OR	ON	2008/02/05 21:53
S57	17	S53 and fence and (vector scalar)	USPAT	OR	ON	2008/02/05 21:53

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1550	(711/151,154,158 712/3,40,225). ccls.	US-PGPUB	OR	ON	2008/02/05 22:11
L2	0	L1 and lsync.clm.	US-PGPUB	OR	ON	2008/02/05 22:11
L3	5	L1 and sync.clm.	US-PGPUB	OR	ON	2008/02/05 22:11
L4	0	L1 and (prevent$4 near2 subsequent near2 scalar).clm.	US-PGPUB	OR	ON	2008/02/05 22:12
L5	0	L1 and (prevent$4 near2 subsequent near2 scalar)	US-PGPUB	OR	ON	2008/02/05 22:12
L6	1	L1 and (vector with invalidat$4 with cache).clm.	US-PGPUB	OR	ON	2008/02/05 22:13
L7	0	L1 and (synchronization adj marker).clm.	US-PGPUB	OR	ON	2008/02/05 22:13
L8	0	L1 and (synchronization adj marker)	US-PGPUB	OR	ON	2008/02/05 22:13

IEEE Xplore® RELEASE 2.5

Welcome United States Patent and Trademark Office

| BROWSE | SEARCH | IEEE XPLORE GUIDE | SUPPORT |

☐ Search Session History

Tue, 5 Feb 2008, 10:07:21 PM EST

Search Query Display

Edit an existing query or compose a new query in the Search Query Display.

Select a search number (#) to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

Recent Search Queries   **Results**

#1	((membar)<in>metadata)	0
#2	((memory barrier)<in>metadata)	5
#3	((memory barrier)<in>metadata)	5
#4	((lsync)<in>metadata)	0

Help   Contact Us   Privacy & Security   IEEE.org

Indexed by
Inspec°

2/5/2008

IP.com
PriorArtDatabase

USPTO

February 05, 2008

Search

Full Text

Concept

Document ID

Recent Disclosures

Other

Prior Art Home

Support

Logout

We are experiencing some temporary problems with our email system. As a result, you may not receive immediate notifications of purchases and publishing submissions. We apologize for the inconvience. If you need immediate assistance, please call Customer Service at 585-427-8180.

Displaying records #1 through 2 out of 2

Result # 1   Relevance: ○○○○○

The Sync Model: A Parallel Execution Method for Logic Programming

1986-03-31   IPCOM000147867D   English (United States)

The Sync Model: A Parallel Execution Method for Logic Programming Pey-yun Peggy Liand Alain J. MartinComputer ScienceCalifornia Institute of Technology Pasadena CA 91125 March 1986 Abstract The Sync Model, a parallel execution method for logic programming, is ...

Result # 2   Relevance: ○○○

SUBMICRON SYSTEMS ARCHITECTURE Semiannual Technical Report

1986-12-05   IPCOM000147935D   English (United States)

Department of Computer Science California hstitu te of Technology 5235:TR:86 5 December 1986 Reporting Period: 16 March 1986 to 15 November 1986 Principal Investigator: Charles L Seitz Faculty Investigators: James T Kajiya Alain J Martin Robert J ...

Displaying page 1 of 1   << FIRST | < BACK | NEXT > | LAST >>

Search query: lsync

New search | Modify this search | Search within current results

IP.com
PriorArtDatabase

Securing innovation

February 05, 2008   USPTO

Search

Full Text

Concept

Document ID

Recent Disclosures

Other

Prior Art Home

Support

Logout

We are experiencing some temporary problems with our email system. As a result, you may not receive immediate notifications of purchases and publishing submissions. We apologize for the inconvience. If you need immediate assistance, please call Customer Service at 585-427-8180.

Displaying records #1 through 3 out of 3

Result # 1   Relevance: ⊙⊙⊙⊙⊙

COMPILING LISP FOR EVALUATION ON A TIGHTLY COUPLED MULTIPROCESSOR

1986-03-20   IPCOM000151501D   English (United States)

iii Abstract The problem of compiling Lisp for efficient evaluation on a large, tightly coupled, shared merriory multiprocessor is investigated. A representation for s-expressions which facilitates parallel evaluation is proposed, along with a sequence of ...

Result # 2   Relevance: ⊙⊙

MINIMUM NORM SOLUTIONS OF SINGLE STIFF LINEAR ANALYTIC DIFFERENTIAL EQUATIONS

1899-12-30   IPCOM000150236D   English (United States)

Inst f6r Informationsbehzndling D e t of Information ProcessS.no o KTH The Xoyal Tnstitute of Technc~iugy 130 44 STOCKIJOLI.I 70 S-I03 44 STOCKHOLM 70, Swede2n MINIMUM NORM SOLUTIONS OF SINGLE STIFF LINEAR ANALYTC DIFFERENTIAL EQUATIONS by Ilkka Karasalo ...

Result # 3   Relevance: ⊙

GENERATIVE, DESCRIPTIVE, FORMAL AND HEURISTIC MODELING IN PATTERN ANALYSIS AND CLASSIFICATION

1971-03-31   IPCOM000151570D   English (United States)

Technical Report TR-151 AFOSR71-1982 March 1971 GENERATIVE, DESCRIPTIVE, FORMAL AND HEURISTIC MODELING IN PATTERN ANALYSIS AND CLASSIFICATION This research was supported in part by the Mathematics and Information Sciences Directorate, Air Force Office of ...

Displaying page 1 of 1   << FIRST | < BACK | NEXT > | LAST >>

Search query: fence AND vector AND scalar

New search | Modify this search | Search within current results

⊙ip.com™

IP.com PriorArtDatabase

Securing innovation

February 05, 2008

USPTO

Search

Full Text

Concept

Document ID

Recent Disclosures

Other

Prior Art Home

Support

Logout

We are experiencing some temporary problems with our email system. As a result, you may not receive immediate notifications of purchases and publishing submissions. We apologize for the inconvience. If you need immediate assistance, please call Customer Service at 585-427-8180.

Displaying records #1 through 2 out of 2

Result # 1   Relevance: ⊙⊙⊙⊙⊙
DASD Availability Topics
1985-05-31   IPCOM000148581D   English (United States)
J. R. Garner GG66-0202 May, 1985 The information contained in this document has not been submitted to any formal IBM test and is distributed on an "as is" basis without any warranty either expressed or implied. The use of this information or the ...

Result # 2   Relevance: ⊙⊙
Operating System Considerations for Large-Scale MIMD Machines
1985-12-31   IPCOM000128239D   English (United States)
In order to realize the potential of highly parallel shared-memory MIZv1D architectures for solving very large problems, novel challenges must be faced by the system software designer. The operating system must endeavor to utilize all processors fully, without incurring ...

Displaying page 1 of 1   << FIRST | < BACK | NEXT > | LAST >>

Search query: fence AND sync

New search | Modify this search | Search within current results

IP.com
PriorArtDatabase

ip.com™

Securing *innovation*

USPTO

Search

Full Text

Concept

Document ID

Recent Disclosures

Other

Prior Art Home

Support

Logout

Displaying records #1 through 10 out of 13

Result # 1   Relevance: ⬤⬤⬤⬤⬤

Compiling Ordinary Programs for Execution on an Asynchronous Multiprocessor

1899-12-30   IPCOM000148160D   English (United States)

A Dissertationby Thomas L. Rodeheffer Although the asynchronous multiprocessor can perform several simultaneous sequences of actions, its separate processors do not execute in unison, but require explicit operations to achieve coordination, An optimizing compiler for such ...

Result # 2   Relevance: ⬤⬤⬤◯

Preventing the concurrent collection from tracing into an unassigned local copy of an object on weak ordering hardware

2002-07-01   IPCOM000015822D   English (United States)

Disclosed is a method for allowing concurrent garbage collectors, running on a weak ordering hardware, to trace only into objects whose assigned copy was already made global. The problem solved by this method: Garbage collectors designed for large server configurations ...

Result # 3   Relevance: ⬤⬤◯

COMPILING LISP FOR EVALUATION ON A TIGHTLY COUPLED MULTIPROCESSOR

1986-03-20   IPCOM000151501D   English (United States)

iii Abstract The problem of compiling Lisp for efficient evaluation on a large, tightly coupled, shared merriory multiprocessor is investigated. A representation for s-expressions which facilitates parallel evaluation is proposed, along with a sequence of ...

Result # 4   Relevance: ⬤⬤◯

Considerations for Massively Parallel UNIX Systems on the NYU Ultracomputer and IBM RP31

1985-12-31   IPCOM000128238D   English (United States)

Novel challenges must be met when designing UNIX implementations for highly parallel shared-memory MIIVID architectures. Of primary importance is the need to avoid serial bottlenecks whenever possible, so that the potential speedup of such machines can be realized. ...

Result # 5   Relevance: ◯◯

Operating System Considerations for Large-Scale MIMD Machines

1985-12-31   English (United States)

IPCOM000128239D

In order to realize the potential of highly parallel shared-memory MIZv1D architectures for solving very large problems, novel challenges must be faced by the system software designer. The operating system must endeavor to utilize all processors fully, without incurring ...

Result # 6   Relevance: ◯◯

DASD Availability Topics

1985-05-31   English (United States)

IPCOM000148581D

J. R. Garner GG66-0202 May, 1985 The information contained in this document has not been submitted to any formal IBM test and is distributed on an "as is" basis without any warranty either expressed or implied. The use of this information or the ...

Result # 7   Relevance: ◯

enterprise systems connection (ESCON) architecture-system overview

1992-07-31   English (United States)

IPCOM000163900D

Enterprise Systems Connection (ESCON) Architecture- System by S. A. Calta J. A. deVeer E. Loizides R. N. Strangwayes overview This paper serves as an introduction to a wholly new IBM data processing interconnection system called Enterprise Systems Connection ...

Result # 8   Relevance: ◯

Evolution of the DASD storage control

1989-06-30   English (United States)

IPCOM000165091D

This paper identifies the major requirements and de- sign points for storage controls and describes how these requirements have been met over time. It also describes the interplay of the three critical compo- nents of a subsystem: hardware technology, micro- code, and ...

Result # 9   Relevance: ◯

Method for a nonblocking copy engine using SMT and SOEMT threads

04-Jun-2004   English (United States)

IPCOM000028868D

Disclosed is a method for a nonblocking copy engine using simultaneous multithreading (SMT) and switch-on-event multithreading (SOEMT) threads. Benefits include improved functionality and improved performance.

Result # 10   Relevance: ◯

Initial Report on a Lisp Programmer's Apprentice

1976-12-31   English (United States)

IPCOM000149196D

SECURITY CLASSIFICATION OF THIS PAGE (man Dola Enleesd) Initial Report on a Lisp Programmer's Technical Report 9. PERFORMING ORGANIZATION NAME AND ADDRESS 10. PROGPAM ELEMENT, PROJECT, TASK AREA 8, WORK UNIT NUMBERS Artificial intelligence Laboratory 545 Technology ...

Displaying page 1 of 2 << FIRST | < BACK | NEXT > | LAST >>

Search query: fence AND synchronization

New search | Modify this search | Search within current results

ip.com

IP.com PriorArtDatabase

Securing innovation

February 05, 2008   USPTO

Search

Full Text

Concept

Document ID

Recent Disclosures

Other

Prior Art Home

Support

Logout

We are experiencing some temporary problems with our email system. As a result, you may not receive immediate notifications of purchases and publishing submissions. We apologize for the inconvience. If you need immediate assistance, please call Customer Service at 585-427-8180.

Displaying records #11 through 13 out of 13

Result # 11   Relevance: ○

instruction scheduling in the TOBEY compiler

1994-09-30   IPCOM000164037D   English (United States)

The high performance of pipelined, superscalar processors such as the POWERS" and PowerPC" is achieved in large part through the parallel execution of instructions. This fine-grain parallelism cannot alwaysbe achieved by the processor alone, butrelies to some ...

Result # 12   Relevance: ○

Applicability of Remote Direct Memory Access Protocol (RDMA) and Direct Data Placement (DDP) (RFC5045)

2007-10-01   IPCOM000159873D   English (United States)

This document describes the applicability of Remote Direct Memory Access Protocol (RDMAP) and the Direct Data Placement Protocol (DDP). It compares and contrasts the different transport options over IP that DDP can use, provides guidance to ULP developers on choosing between ...

Result # 13   Relevance: ○

DYNAMIC ANALYSIS OF EXECUTION: Possibilities, Techniques and Problems

1973-09-30   IPCOM000147958D   English (United States)

DYNAMIC ANALYSIS OF EXECUTION: Possibilities, Techniques and Problems b Y Birol her Aygiin Department of Computer Science Carnegie-Mellon University Pittsburgh, Pennsylvania 15213 September, 1973 This work was supported by the Advanced Research Projects ...

Displaying page 2 of 2   <<FIRST | <BACK | NEXT > | LAST >>

Search query: fence AND synchronization

New search | Modify this search | Search within current results

Google

membar	Search	Advanced Search
		Preferences

Web

Results **1 - 10** of about **34,800** for **membar**. (0.05 seconds)

SIPExpressRouter: atomic/atomic_alpha.h File Reference
#define, **membar**() asm volatile (" mb \n\t" : : : "memory"). #define, membar_read()
membar(.... Value:. do{ \ **membar**(); \ atomic_and_int(v, m); \ }while(0) ...
siprouter.onsip.org/doc/doxygen/atomic__alpha_8h.html - 88k - Cached - Similar pages

[PATCH] membar for sparc64
[PATCH] **membar** for sparc64. To: libc-hacker at sourceware dot cygnus dot com ...
+#define MEMORY_BARRIER() __asm__ __volatile__("**membar** 9" : : : "memory") ...
sourceware.org/ml/libc-hacker/2000-04/msg00078.html - 5k - Cached - Similar pages

Bug ID: 4827353 atomic::membar doesn't on x86
hotspot:runtime_system, atomic::**membar** doesn't on x86,State: closed,Reported: 05-MAR-
2003,Release Reported Against: mantis-beta.
bugs.sun.com/bugdatabase/view_bug.do?bug_id=4827353 - 20k - Cached - Similar pages

Aldy Hernandez - patch: [FRV] membar optimization
While looking at uncontributed patches we have, I noticed we had a much better __builtin_
{read,write}* implementation that removes redundant **membar** ...
gcc.gnu.org/ml/gcc-patches/2005-07/msg01857.html - 39k - Cached - Similar pages

Processor memory unit
A SPARC V9 **MEMBAR** can specify one or more of several ordering options. ... **MEMBARs**
are ignored under the processor consistency and total store ordering ...
rsim.cs.uiuc.edu/rsim/Manual/node25.html - 15k - Cached - Similar pages

(WO/2005/121948) METHOD AND APPARATUS FOR ENFORCING MEMBAR ...
A method for executing a **membar** instruction in an execute-ahead processor, wherein the
membar instruction forces buffered loads and stores to complete ...
www.wipo.int/pctdb/en/wo.jsp?IA=WO2005121948&DISPLAY=CLAIMS - 20k -
Cached - Similar pages

[PDF] CS252 Graduate Computer Architecture Memory Models Practice Problems
File Format: PDF/Adobe Acrobat - View as HTML
The system has a **MEMBAR** memory barrier instruction that guarantees the effects of. all
memory instructions executed before the **MEMBAR** will be made globally ...
inst.eecs.berkeley.edu/~cs252/fa07/handouts/CS252-MemoryModels-Practice.pdf -
Similar pages

Linux-Kernel Archive: [PATCH][3/8] Arch agnostic completely out of ...
+ __asm__ __volatile__ ("**membar** #LoadStore | #StoreStore\n\t" + "retl\n\t" .... **membar**
#LoadLoad - ba,pt %xcc, 1b ! Retry lock acquire - wrpr %g2, %pil ! ...
www.ussg.indiana.edu/hypermail/linux/kernel/0409.0/0735.html - 19k -
Cached - Similar pages

SPARC64-III User's Guide Errata (V2.0) Feb. 16 1999 *** This file ...
The ordering of **MEMBAR** instruction itself is always maintained in order before and after all
kinds of memory instruction in all memory models. ...
www.sparc.org/standards/sparc64.errata.txt - 19k - Cached - Similar pages

---

#$Id: sysinfo.pl,v 1.3 2001/05/23 13:16:52 where Exp $ #!/usr/bin ...
... chop ($MODEL); #--SPEW IT INTO THE CHANNEL SHALL WE--# IRC::command
("SysInfo: $UNAME | $MODEL $CPU MHz | Mem: $MEMFREE/$MEMTOTAL **$MEMBAR** |
Diskspace: ...
dragoncat.net/lists/irssi-users/2002-02/att-0049/01-sysinfo.pl - 7k - Cached - Similar pages

1 2 3 4 5 6 7 8 9 10   **Next**

| membar | Search |

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve | Try Google Experimental

©2008 Google - Google Home - Advertising Programs - Business Solutions - About Google

Google

| membar instruction | Search | Advanced Search Preferences |

Web   Results **1 - 10** of about **5,420** for **membar** instruction. (**0.23** seconds)

Did you mean: ***member*** instruction

(WO/2005/121948) METHOD AND APPARATUS FOR ENFORCING **MEMBAR** ...
A method for executing a **membar instruction** in an execute-ahead processor, ... The
method of claim 1, wherein when the **membar instruction** is deferred, ...
www.wipo.int/pctdb/en/wo.jsp?IA=WO2005121948&DISPLAY=CLAIMS - 20k -
Cached - Similar pages

(WO/2005/121948) METHOD AND APPARATUS FOR ENFORCING **MEMBAR** ...
For example, one common **membar instruction** forces a deterministic memory state ...
Upon encountering a **membar instruction**, the processor determines if the ...
www.wipo.int/pctdb/en/wo.jsp?IA=WO2005121948&DISPLAY=DESC - 38k -
Cached - Similar pages

[PDF] CS252 Graduate Computer Architecture Memory Models Practice Problems
File Format: PDF/Adobe Acrobat - View as HTML
The system has a **MEMBAR** memory barrier **instruction** that guarantees the effects ...
There is no generalized **MEMBAR instruction** as in Part B of this problem. ...
inst.eecs.berkeley.edu/~cs252/fa07/handouts/CS252-MemoryModels-Practice.pdf -
Similar pages

SPARC64-III User's Guide Errata (V2.0) Feb. 16 1999 *** This file ...
Therefore any memory prefetch ordering by PREFETCH(A) **instruction** is not guaranteed in
any memory model. - The ordering of **MEMBAR instruction** itself is ...
www.sparc.org/standards/sparc64.errata.txt - 19k - Cached - Similar pages

System and method for parallel execution of memory transactions ...
For the purposes of this document, five types of **membar instructions** are defined: ...
membar load-load: all load **instructions** before this **membar instruction** ...
www.patentstorm.us/patents/5893165-description.html - 61k - Cached - Similar pages

Aldy Hernandez - patch: [FRV] **membar** optimization
+ next_is_end_p = true; + + /* If the current **instruction** is a __builtin_read or
__builtin_write, + NEXT_MEMBAR is the **membar instruction** associated with it ...
gcc.gnu.org/ml/gcc-patches/2005-07/msg01857.html - 39k - Cached - Similar pages

(WO/2005/121948) METHOD AND APPARATUS FOR ENFORCING **MEMBAR** ...
Upon encountering a **membar instruction**, the processor determines if the load buffer
and ... If so, the processor defers the **membar instruction** and executes ...
www.wipo.org/pctdb/en/wo.jsp?wo=2005121948 - 15k - Cached - Similar pages

System and method for parallelizing execution of memory ...
membar memsync: all memory transaction **instructions** (of all types) before ..... The
membar override logic 152 works as follows. If a **membar instruction** is ...
www.freepatentsonline.com/EP0817091.html - 70k - Cached - Similar pages

Asymmetric Dekker Synchronization ...
The **MEMBAR instruction** directs the processor to make the effect of the ST visible ...
Unfortunately the **MEMBAR instruction** is costly on SPARC anid IA32 ...

blogs.sun.com/dave/resource/Asymmetric-Dekker-Synchronization.txt - 129k -
Cached - Similar pages

Method and apparatus for enforcing **membar instruction** semantics in ...
One embodiment of the present invention provides a system that facilitates executing a
memory barrier (**membar**) **instruction** in an execute-ahead processor, ...
www.freshpatents.com/Method-and-apparatus-for-enforcing-**membar-instruction**-
semantics-in-an-execute-ahead-... - 26k - Cached - Similar pages

Did you mean to search for: *member* instruction

1 2 3 4 5 6 7 8 9 10   **Next**

| membar instruction | Search |

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve | Try Google Experimental

©2008 Google - Google Home - Advertising Programs - Business Solutions - About Google